Scientific Research Seminar at HSE

Modern Machine Learning for Music Data

Arsenii Ashuha

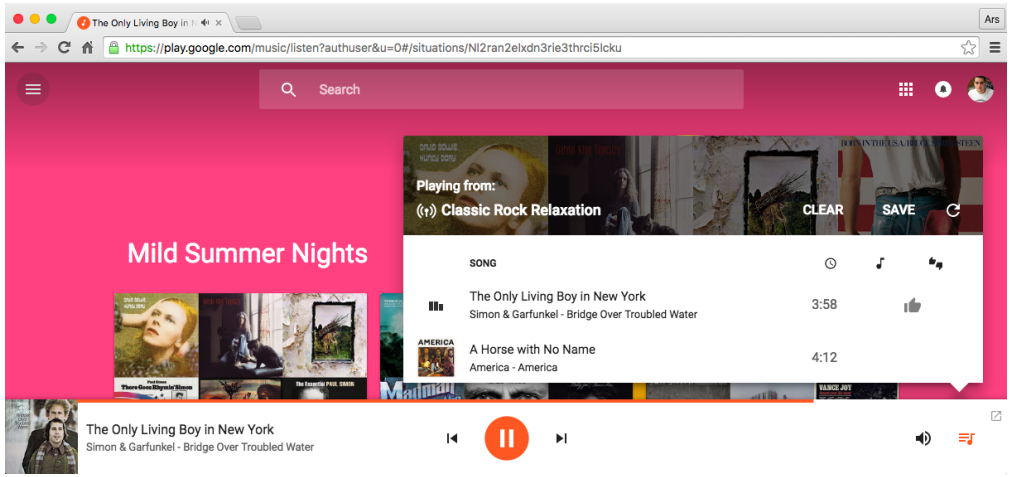
HSE, bayesgroup.ru,

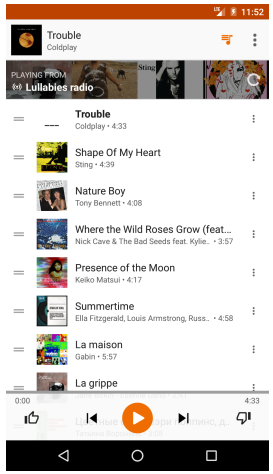
mail: ars.ashuha@gmail.com, **slides**: https://github.com/ars-ashuha/slides

September 12, 2017

How to apply ML for Music Data to get Money?

- Your are working in a big music service as a data scientist





- In this service there's a lot of music data – mp3 files

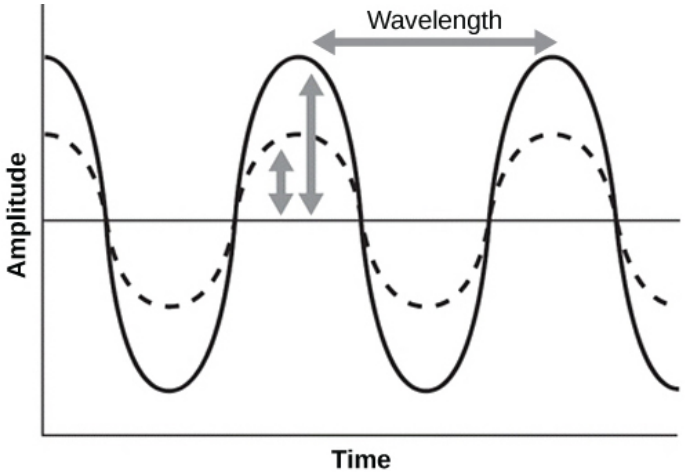
user_id	tracks_id
123	[1, 2, 3]
124	[1000, 11, 23, 23]
...	...
999999	[1]

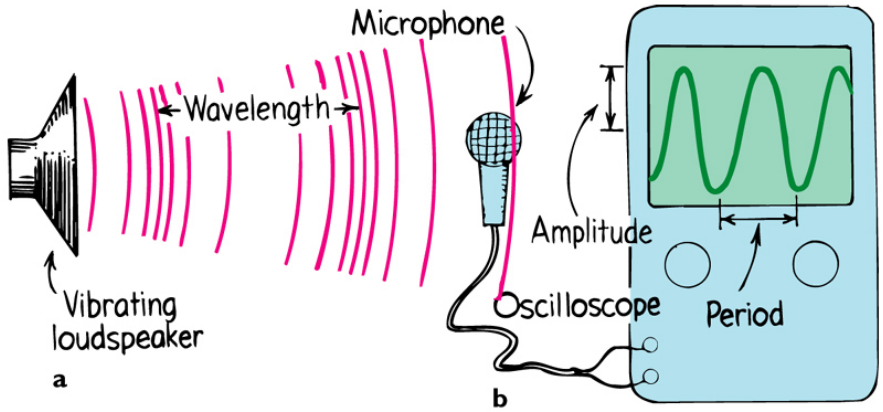
tracks_id	file
1	1.mp3
2	2.mp3
...	...
999999	999999.mp3

- You were given the task – make money using this data

What is the sound?

- Waves and Recording





- How to store sound? Store as big-big array with sampling frequency

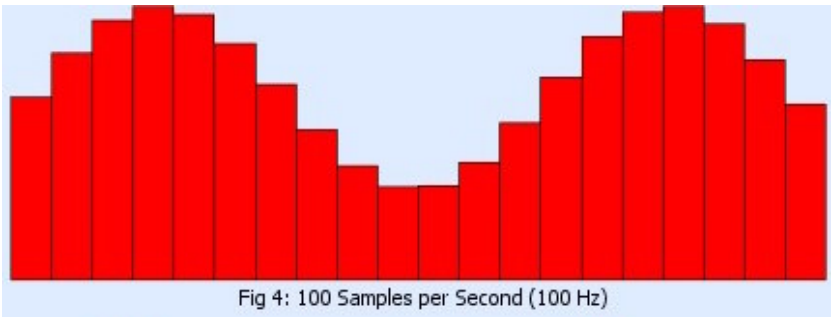


Fig 4: 100 Samples per Second (100 Hz)

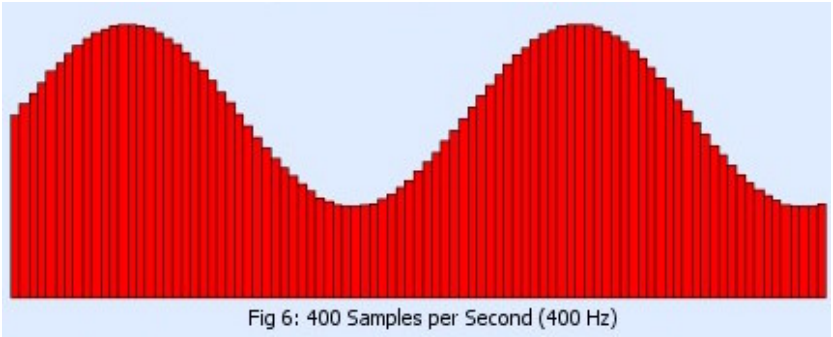


Fig 6: 400 Samples per Second (400 Hz)

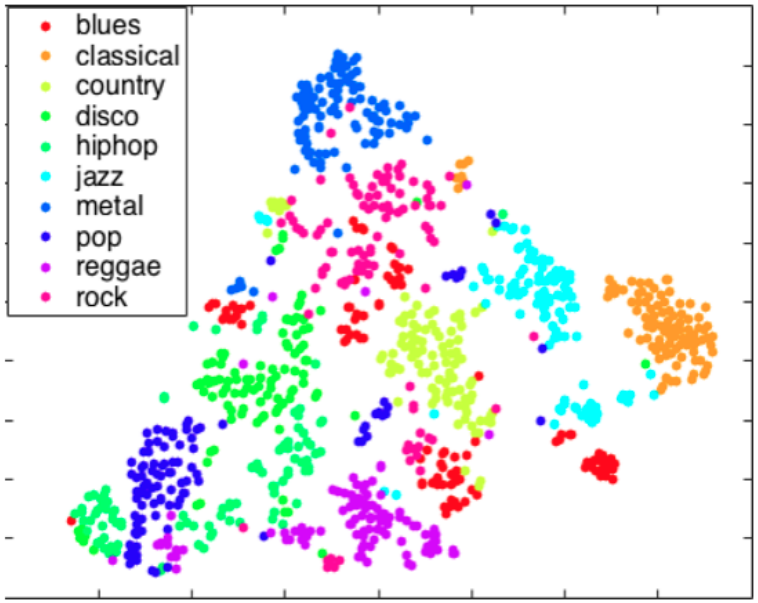
- $[1, 2, 3, 5, 3, 2, 1, 1, 1, 1, 2, 3, 5, 3, 2]$, Usually 16 000 float per second

Finding similar tracks

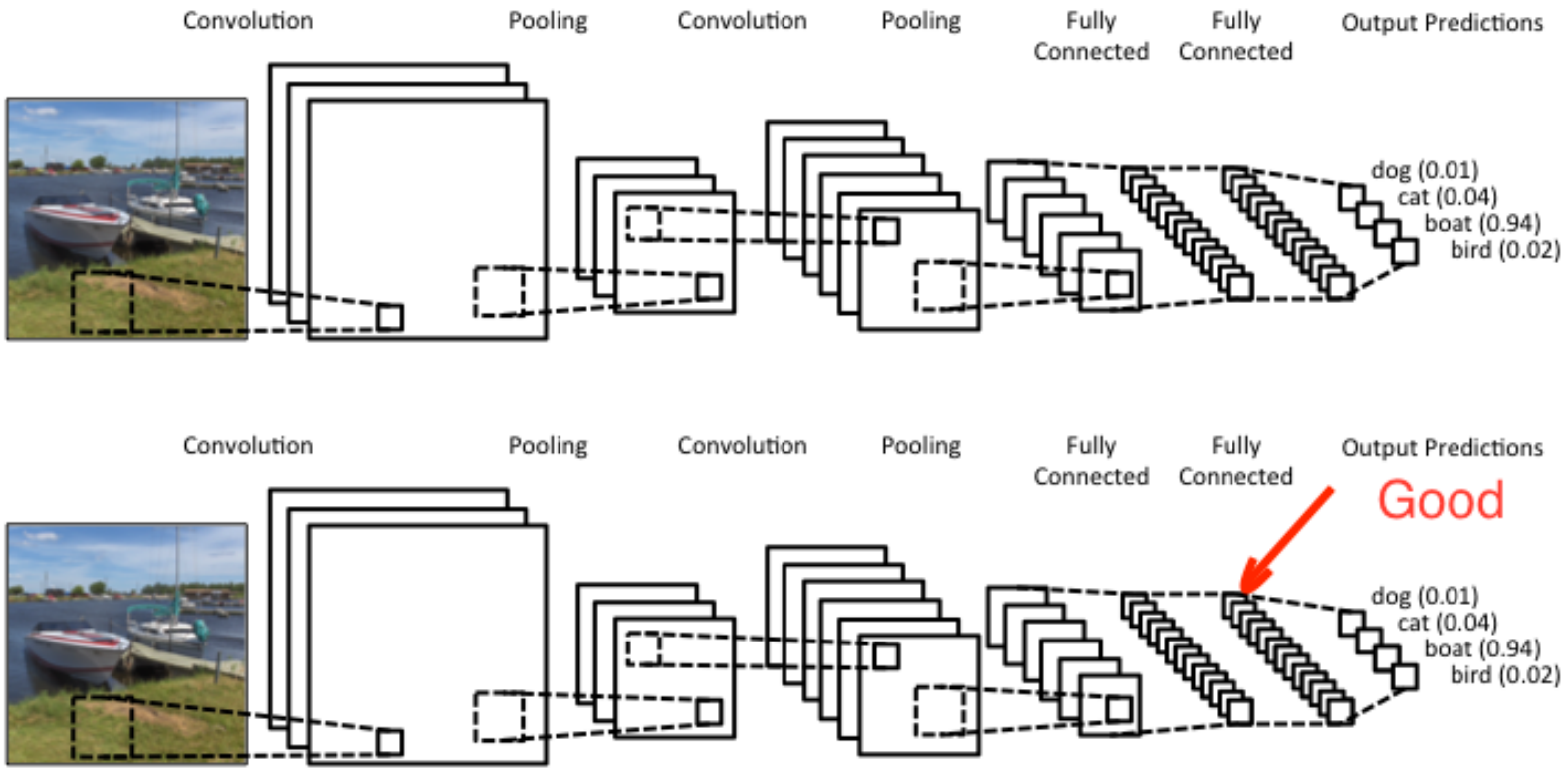
- How to find similar tracks using ML methods?

  Data: 30 sec * 16000 features, 10^7 items

  Task: define function of $similarity(track_i, track_j)$
- Ordinary methods are bad: shift and noise tolerance, over-fitting
- Metric approach is still good idea, if we have a high level description
- Good representation of music track
  - Human – guitar, rock, Queen, 1997, UK, 3 min., ...
  - Computer – good small vector of numbers



Get good representation using Neural Nets

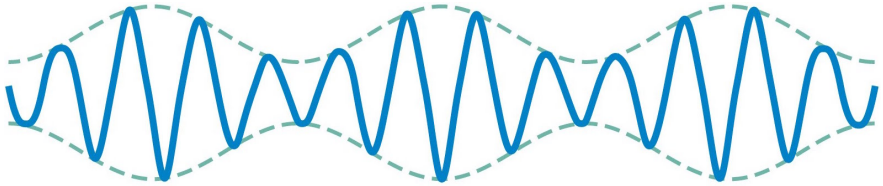


Problem

We need to have picture!

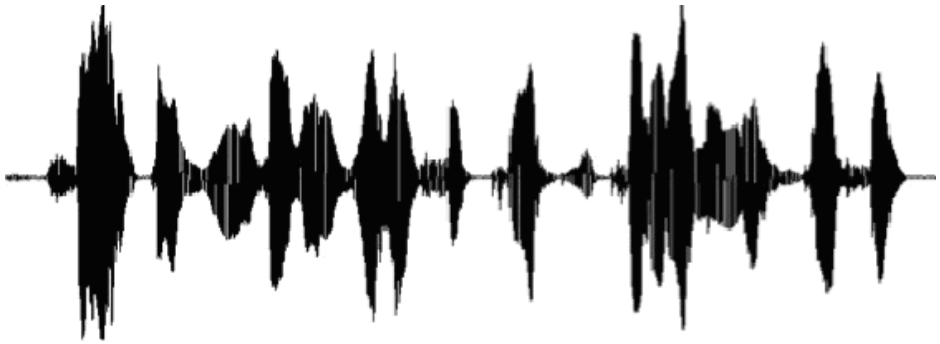
What is the sound part 2?

We have some wave



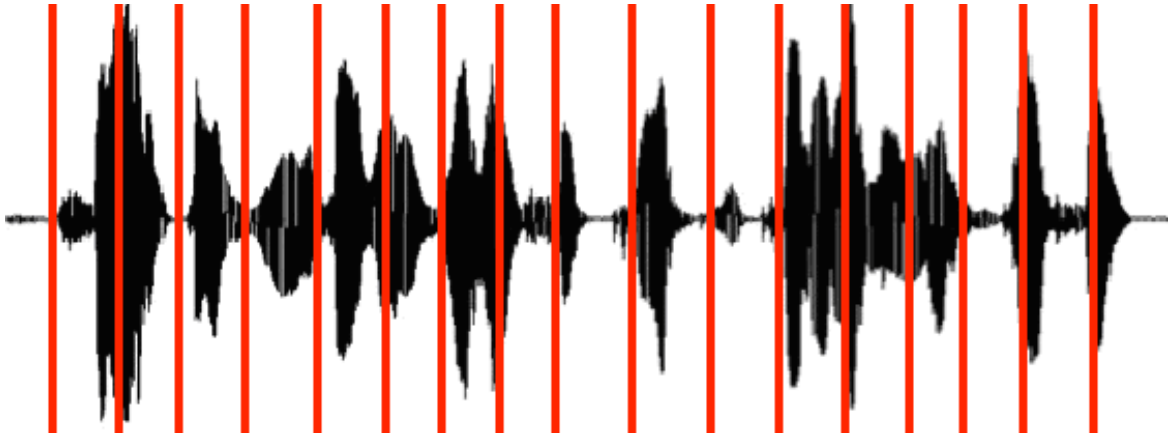
represent wave as a sum of two waves

sound is a combination of big wave range



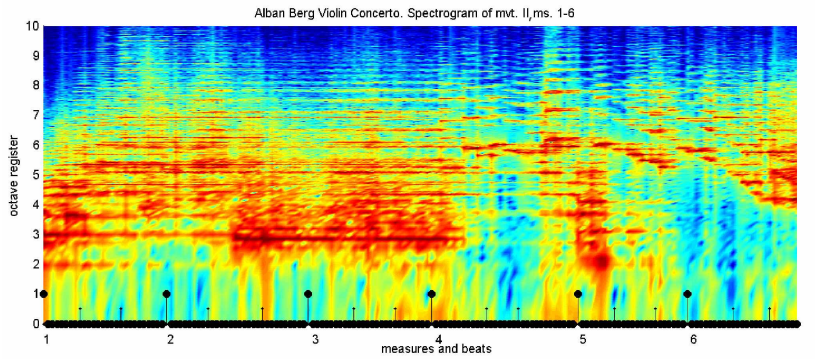
What we lost in our representation?

What is the sound part 2? Get Frequency



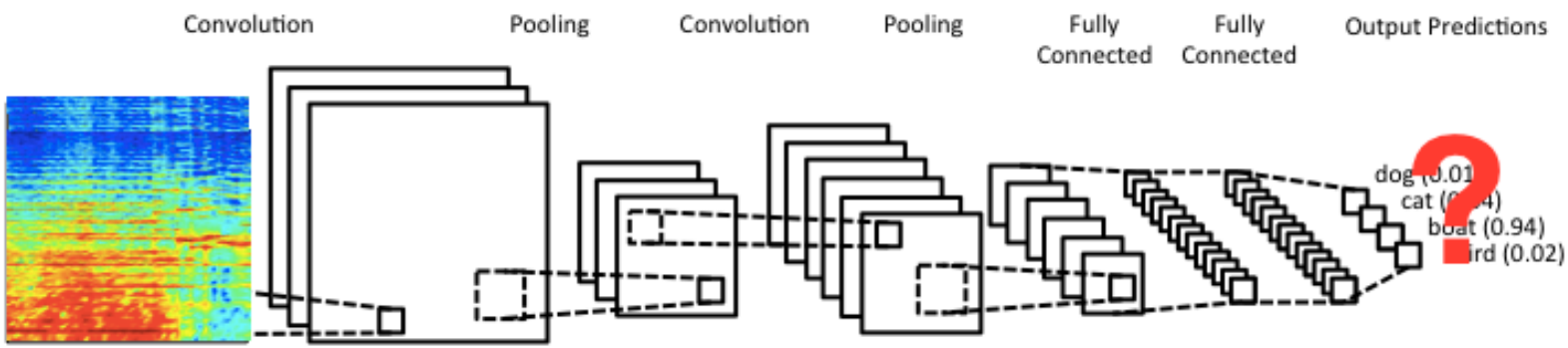
High Freq	1	2	1	1	2	1	1	2	1	1	2	1	1	2	1	1	2	1
Mid Freq	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2
Low Freq	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2

High Freq	1	2	1	1	2	1	1	2	1	1	2	1	1	2	1	1	2	1
Mid Freq	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2
Low Freq	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2	1	2	2



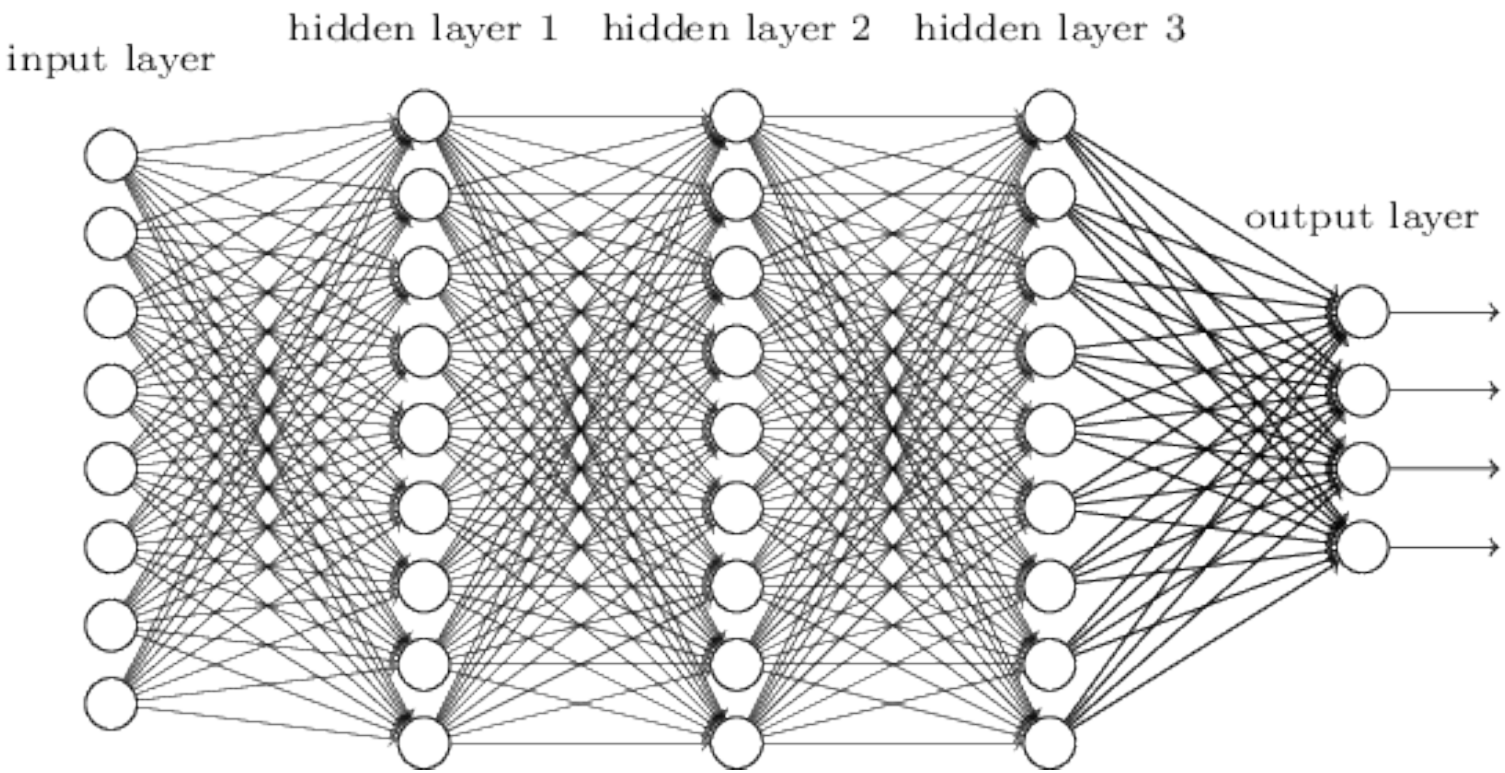
We need to train Neural Nets, but how can we do that?!

- But how can we train nets on music?
- Let's invent a fake machine learning task



  - genre classification
  - artist classification
  - rating prediction
  - .....

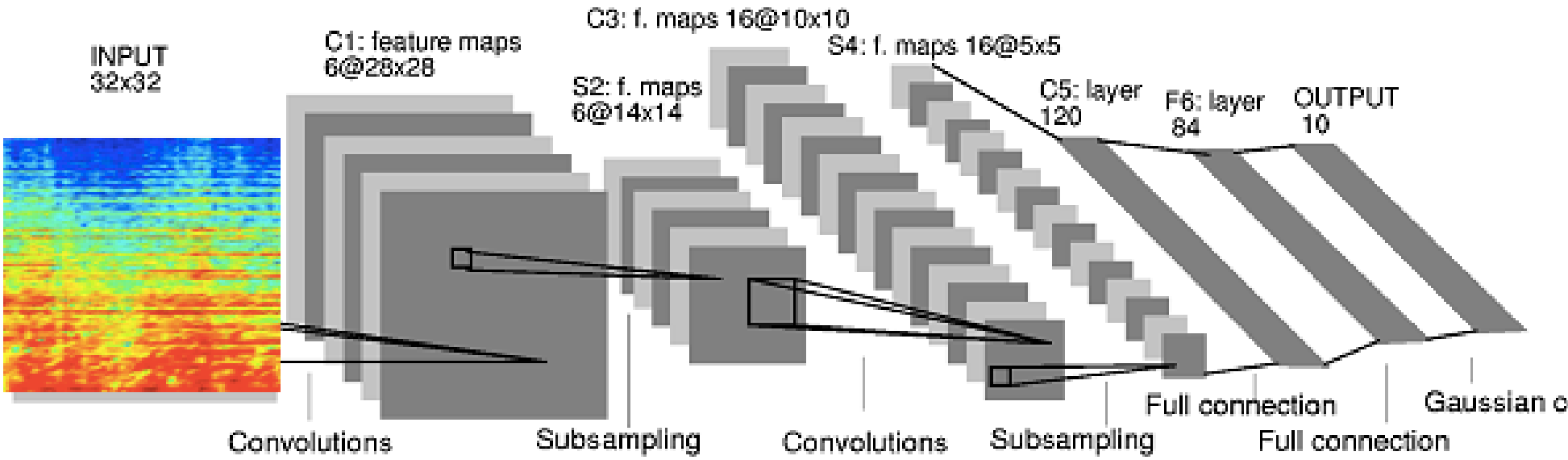
Fully connected NN



- too many parameters – number of weights $= 16^4 * \textit{neurons} + \ldots$
- It doesn't work =(

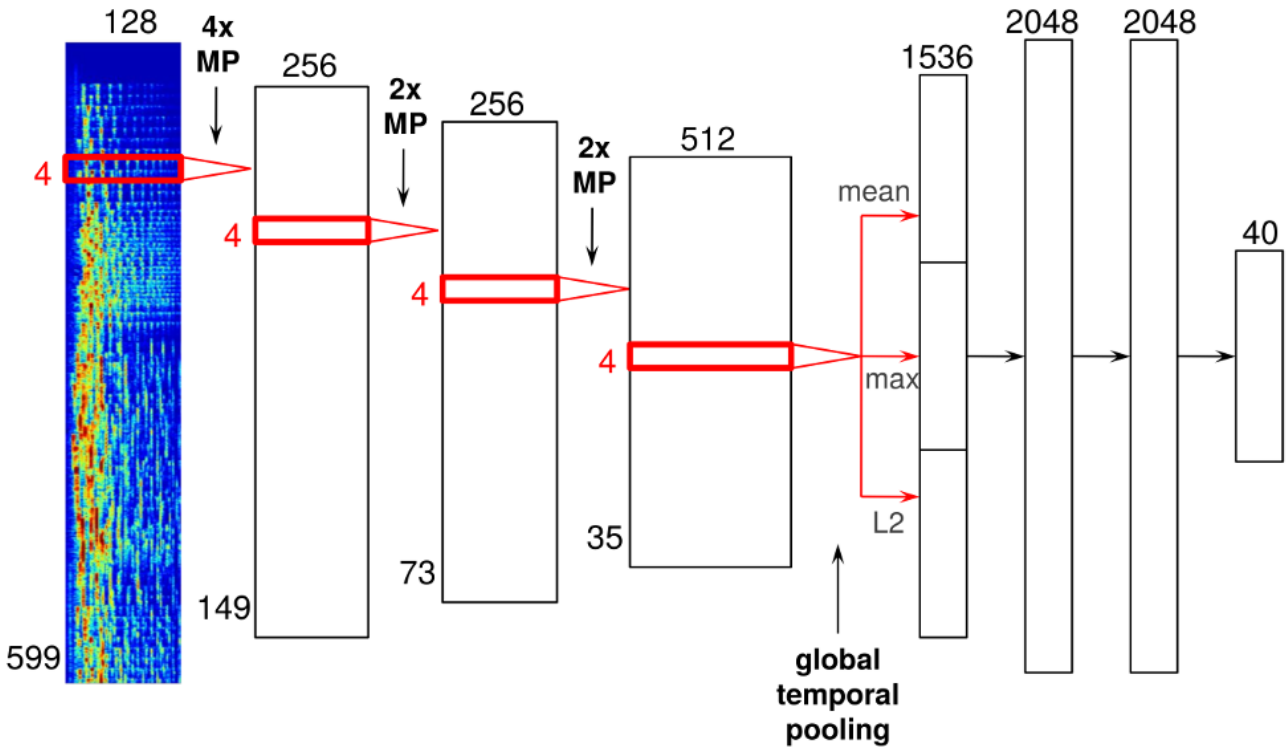
Convolution NN

Is it good for our problem?



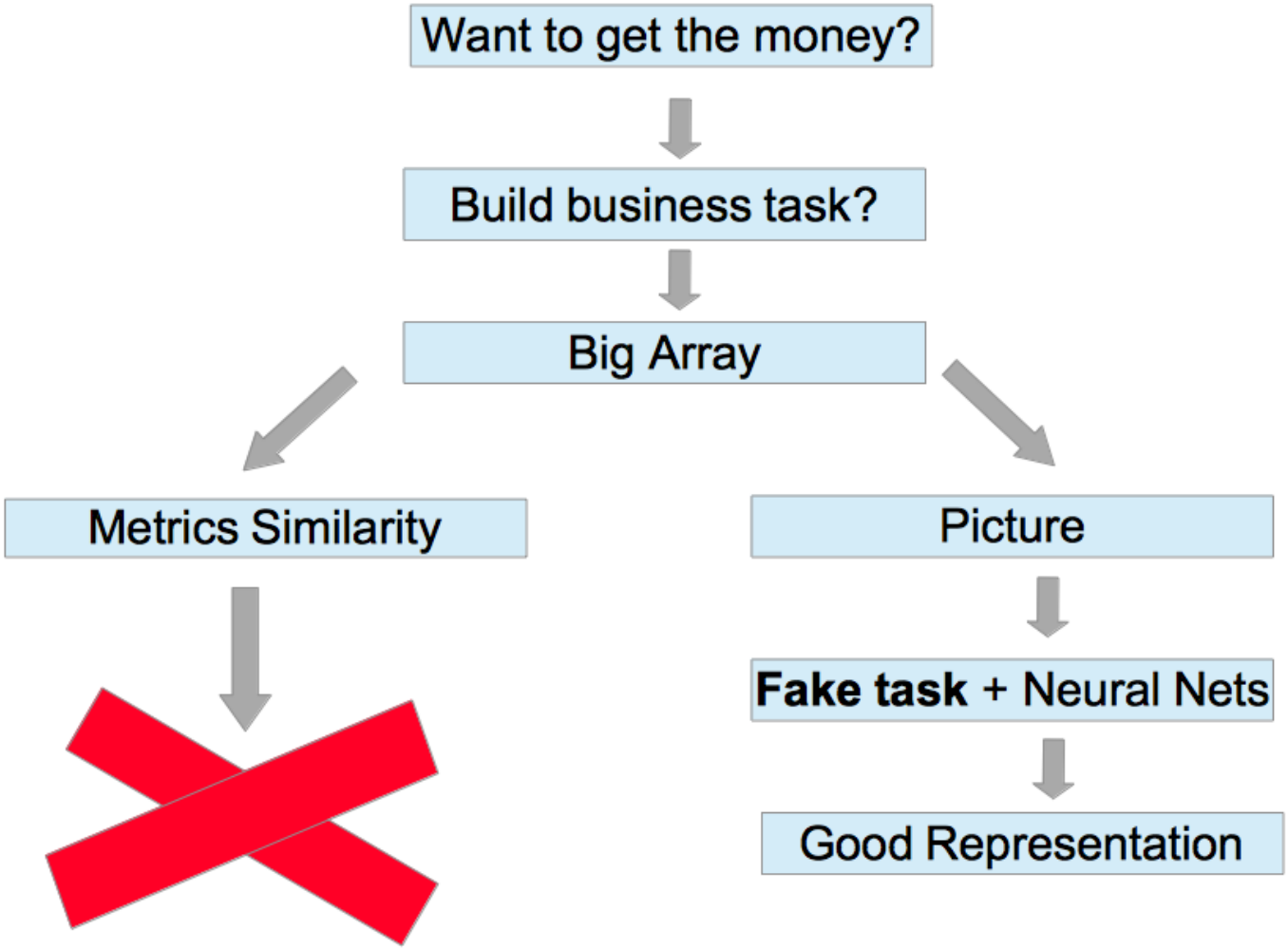
Convolution NN

1D Convolutions



important detail – pooling of time axis [Spotify))) Deep Learning]

General scheme, what did we do?



How to measure quality of good representation?

What we have?

- We have represented each track as a vector
- But maybe our solution is too bad, how can we understand that?
- How to test "good representation"?

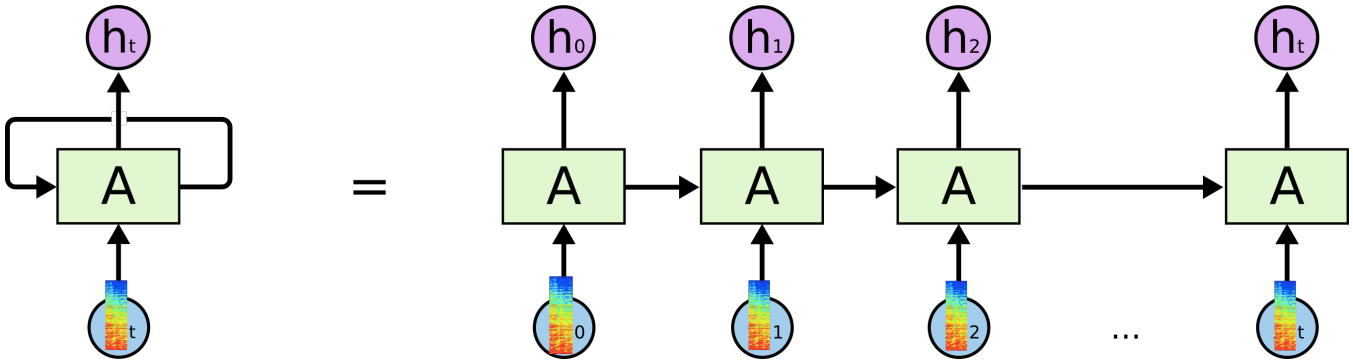
Let's invent the metrics:

- by hand
- using assessors
- recommendation quality
- using vectors to classify another labels

Let's adapt to Different length and Additional information

How to use any length?:

1. Average prediction for many patches
2. Recurrent neural net on many patches



How to take account?:

1. Lyrics

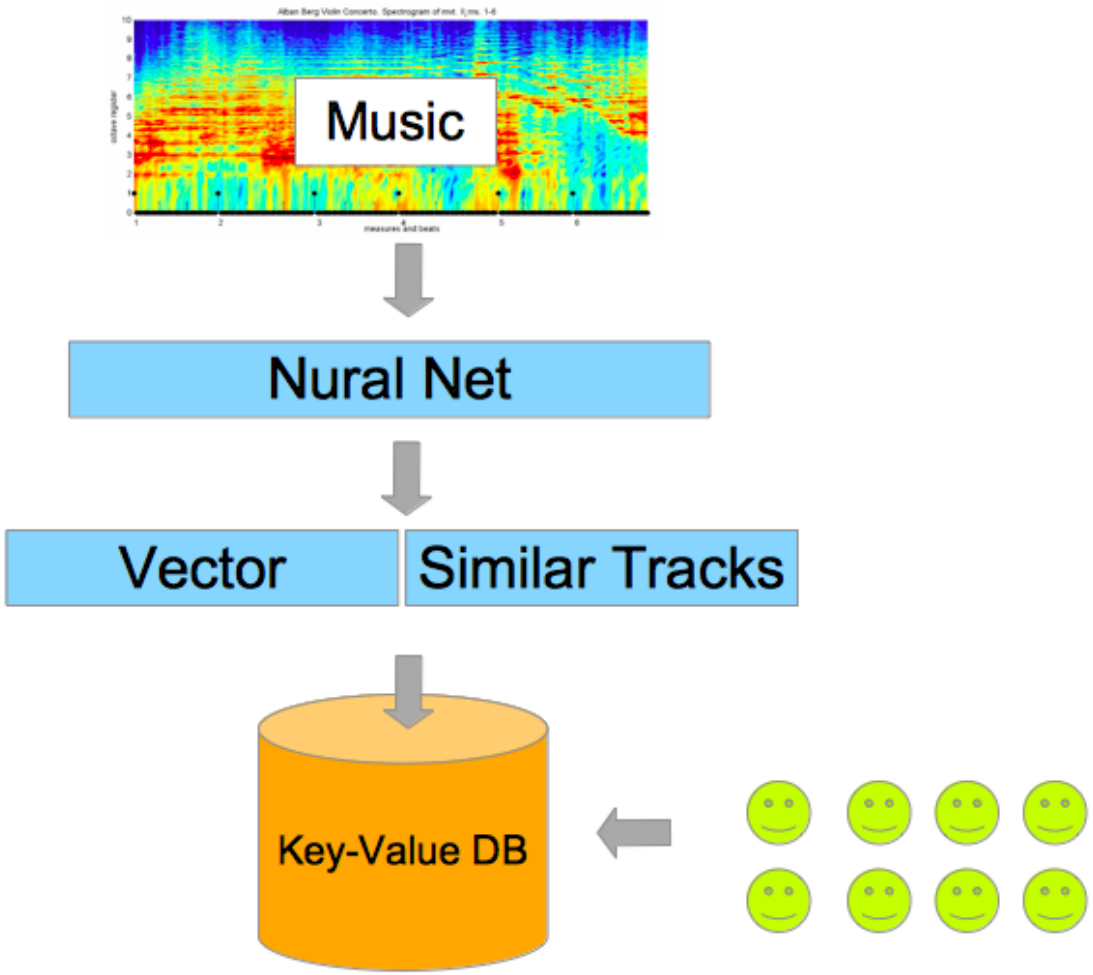
$$\text{Concat}(\text{TextRNN}, \text{Conv}) \rightarrow \text{FC} \rightarrow \text{Cost}$$

2. Genre, Artist, Year – embedding too, multi-cost task

Technical details

How to build fast system for million users?

1. pre-compute vectors and tracks simulation
2. fast key-value storage



End

- Blog post: http://benanne.github.io/2014/08/05/spotify-cnns.html
- NIPS paper: https://papers.nips.cc/paper/5004-deep-content-based-music-recommendation

Current Status of your Field!